Department of Computer Science
University of Hong Kong
Final Year Project

Detailed Project Plan

Author: *Man Tsz Hin*

Date: *Oct 1, 2024*

Table of Contents

1. INTRODUCTION

1.1 Purpose

This plan introduces the software product that will be developed for the Final Year Project (FYP), outlining its functions and the milestones for the entire project. It aims to give classmates and teachers a comprehensive overview of the product and the implementation steps. Detailed implementation information will be provided in the interim report.

1.2 Scope

Learning Intelligent (the app) is an application that aggregates learning materials and utilizes a trained AI based on the materials to create a relaxed and accessible knowledge exchange platform. Unlike traditional platforms that offer structured courses and recognized certifications, Learning Intelligent aims to facilitate peer-to-peer sharing of notes and collaborative learning. It is anticipated that users will perceive Learning Intelligent as a valuable resource for exploring topics of personal interest through contributions from others.

1.3 Overview

This paper will first provide the background of Learning Intelligent and describe it's functions. Subsequently, the development tools utilized will be introduced, concluding with a discussion of the project milestones.

2. GENERAL DESCRIPTION

This section will provide the background of Learning Intelligent by comparing it with other similar products, introducing its functions and analyzing its user characteristics. Notes that these are the basic functions that will be implemented, minor changes are expected and more functions will be added for version updates.

2.1 Product Perspective

Nowadays, people are increasingly focused on learning different skills for both work and personal interests. As a result, platforms like Coursera have been created to offer well-structured courses and recognized certificates. However, these platforms mainly provide formal lessons and often require payment, with materials that are usually text heavy. To address the needs of those seeking knowledge for personal interest or looking for brief, insightful information, Learning Intelligent has been proposed.

2.2 Product Functions

In order to achieve the goal mentioned, Learning Intelligent is equipped with the below functions:
1. Login application
2. Searching
3. View notes
4. Do quiz
5. Post notes
6. Comment
7. Ask question to AI
8. Chat with other users
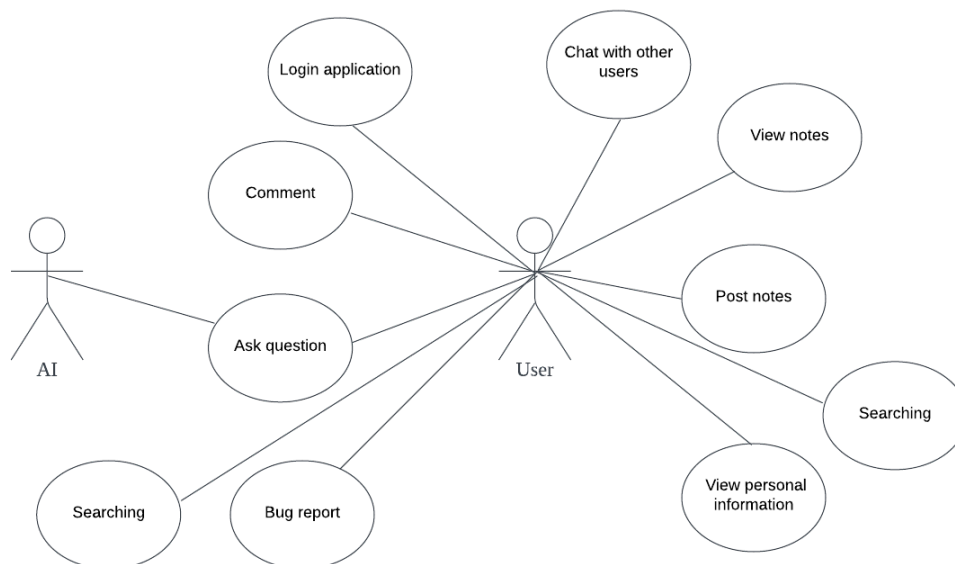9. View personal information
10. Bug report

Figure 1Use case diagram

Each function will be describe in details in Section 3.1

2.3 User Characteristics

The users of Learning Intelligent are primarily individuals motivated by personal interests and a desire for flexible learning options. They include students, professionals seeking to expand their skill sets, and lifelong learners who enjoy exploring new topics outside traditional educational frameworks. These users often prefer informal, interactive learning experiences over structured courses and value the ability to share and access peer-generated content. Additionally, they are likely to appreciate a user-friendly interface that allows for easy navigation and engagement with diverse materials, such as notes, videos, and quizzes. Overall, the Learning Intelligent platform caters to a diverse audience that seeks a collaborative and accessible environment for knowledge exchange.

3. SPECIFIC REQUIREMENTS

This section contains the details of the app's functions, as well as the developing tools that are going to use.

3.1 Functional Requirements

This section specify how the inputs to the software should be transformed into outputs. It describes the fundamental actions that must take place in the software.

3.1.1 Login Application

User will login the application with their account username and password. Once the user clicks the login button, the username and password will be validated. If the credentials are correct, the user will be redirected to the main page. If not, a message box will appear indicating that the username or password is incorrect, allowing the user to close the box and attempt to log in .

3.1.2 Searching

After logging in, users can select notes from the recommendations on the main page or search for topics they are interested in. The app will provide related notes based on matches between the note titles and the search keywords. If there are no matches, the app will respond with a message indicating that no results were found, along with recommendations based on the user's preferences and history. Also, user can search for other user by inputting their username.

3.1.3 View Notes

Once the user opens a note, they can navigate through the content by swiping right for the next page or left for the previous page. The title and description will be displayed at the bottom of the note, and users can scroll down to view them. There may be attachments or links in the description, which users can access by clicking on them. Additionally, users can choose to read the note in full screen or download it to their local device in its original format.

3.1.4 Do Quiz

If a quiz is provided by the poster, there will be a button at the bottom of the note (under the description) and on the last page of the note. Users can click the button to take the quiz. The quiz will be in multiple-choice format, and it will be automatically graded once the user submits it. Users will receive immediate feedback, which will include their score as well as comments from the poster, if any.

3.1.5 Post notes

After logging in, users can post their own notes. They can upload notes in PDF, JPEG, or MP4 formats for others to read directly. Additional formats or extra information can be attached as a link or ZIP file in the description for users to access or download. To post a note, users simply need to fill in the title and description, then upload the note materials. Users can also set a quiz, which will only be in multiple-choice format, during the final step and is optional. For the quiz setup, users need to provide the questions, answer choices, and optional feedback for the results. Finally, just click the "POST" button, and the note will be successfully posted.

3.1.6 Comment

When users are viewing a note, they can choose to either read the description or click the comment button to access the comment section for that note. In the comment section, users can provide feedback on the specific note, such as compliments, corrections, questions about the content, and prompts for further discussion. Human monitoring will be used to ensure that the comments are appropriate.

3.1.7 Ask Question to AI

When users are viewing a note, a button will appear floating in the bottom right corner. When clicked, this button opens a chat box where users can type questions to ask the AI. The corresponding note they are reading will be sent to the AI first, which has been pretrained on the note's content along with some implemented prompts.

3.1.8 Chat with Other User

A chat room function is implemented for communication between users. Users can click on another user's icon to initiate a chat. For example, they can chat with the poster or other users in the comment section by clicking their icons. Additionally, users can directly search for others by entering their usernames, as mentioned in section **3.1.2 Searching**.

3.1.9 View Personal Information

Users can view their own personal information or that of others by clicking the account button or icon, respectively. The personal information includes the user's level and badges, which can be earned by viewing notes, completing quizzes, and posting notes. This gamification system aims to encourage users to engage more actively with the app."

3.1.10 Bug Report

Anyone, regardless of whether they can log in or not, including the general public, can report a bug. When users encounter a bug in our system, they can click the 'Report Bug' button, which will direct them to the bug report page where they can provide details about the issue. The bug report will be stored in the cloud platform.

3.2 Major components

3.2.1 Development Tools

Tool	Description
Flutter	Flutter is an open source framework for building beautiful, natively compiled, multi-platform applications from a single codebase. [1]
Firebase	Firebase is a mobile and web app development platform with a set of cloud tools that helps developers build, deploy and scale their apps. [2]
Poe AI	ChatGPT is an AI language model uses natural language process to generate human-like text responses based on user input, enabling conversational interactions. [3]

3.2.2 Components Interactions

- Users interact with the Flutter app by sending requests.
- Requests are stored in Firebase Firestore for persistence for any changes, and response the information needed
- Firebase will call the chatGPT if needed by the request
- ChatGPT processes the input and returns a response.
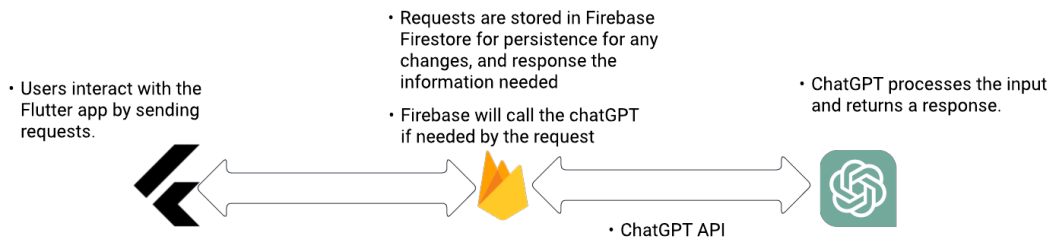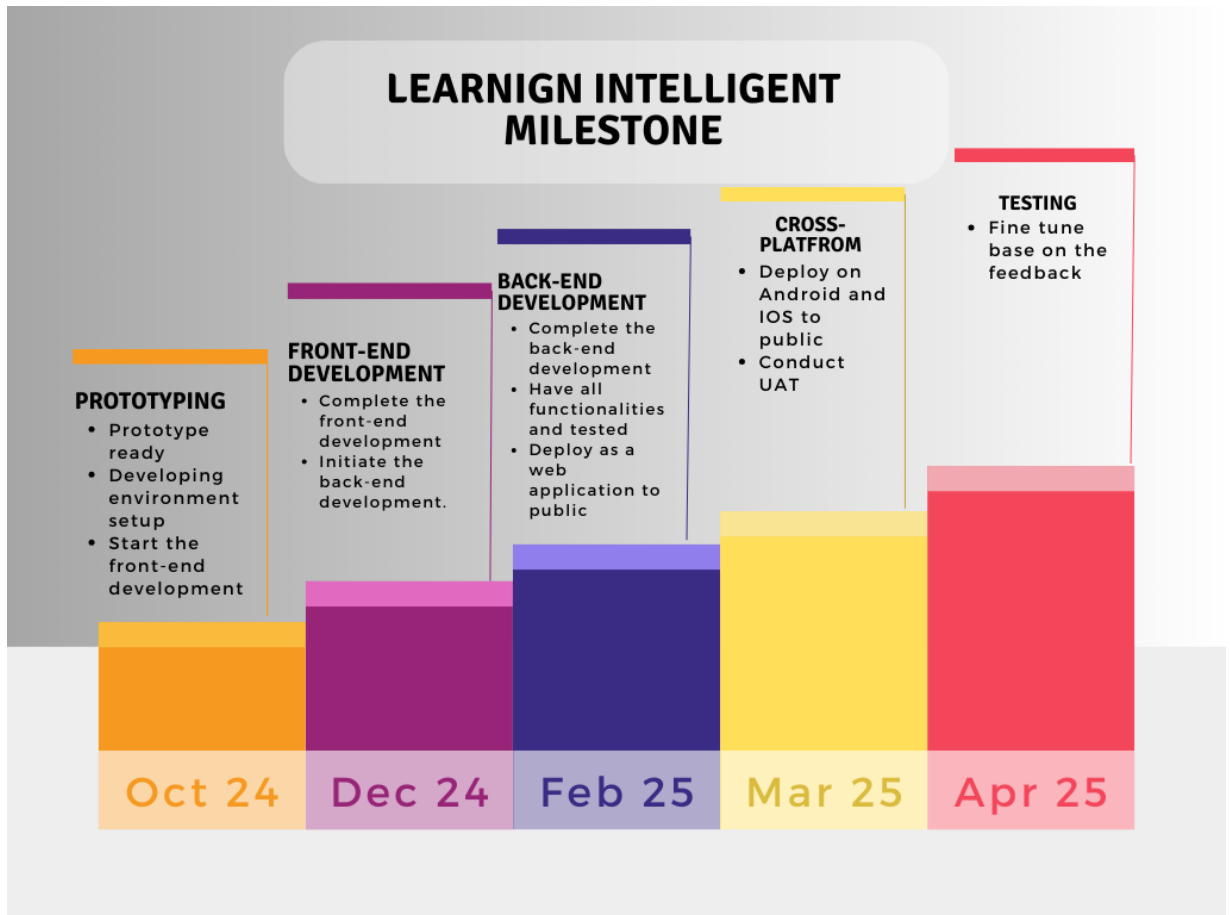- ChatGPT API

Figure 2Flow diagram

Flutter will be utilized for front-end development, encompassing the user interface, state management, and API calls. Firebase will serve as the backend solution, handling database management, server-side logic, and communication with the AI. Finally, Poe AI will be employed by fine-tuning the prompts and training it with the relevant notes each time the user poses a question in the app.

3.3 Testing

To ensure the app is functional and ready for launch, an online mobile app testing tool will be utilized to verify its security and reliability. To assess user acceptance, over 20 users will be invited to participate in User Acceptance Testing (UAT). Modifications will be made based on their feedback, if feasible.

4. Milestone

5. Reference

[1] Flutter, "Flutter - Beautiful native apps in record time," https://flutter.dev/ (accessed Oct. 1, 2024)

[2] Google, "Firebase," https://firebase.google.com/ (accessed Oct. 1, 2024)

[3] OpenAI, "Introducing ChatGPT", https://openai.com/index/chatgpt/ (accessed Oct. 1, 2024)